MULTIMEDIA UNIVERSITY®

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2022/2023 (TERM ID 2210)

ECE1026 ALGORITHMS AND DATA STRUCTURES

(All Sections / Groups)

7 FEBRUARY 2023
9:00 A.M – 11:00 A.M.
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This question paper consists of 8 printed pages including the cover page with 4 questions only.

2. Attempt all FOUR (4) questions. All questions carry equal marks and the distribution of marks is given in the questions.

3. Please print all your answers in the answer booklet provided.

4. State all assumptions clearly.

Question 1

(a) A Finite State Automaton (FSA) is shown in Figure Q1(a).

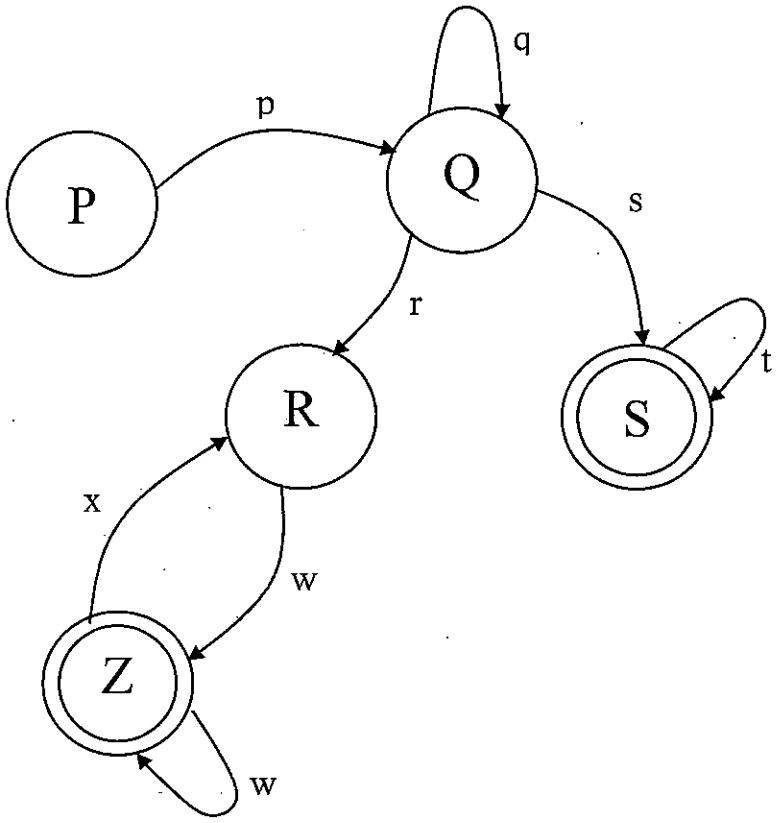

Figure Q1(a)

Based on Figure Q1(a),

(i) Write the Regular Expression (RE) for this FSA.

[7 marks]

(ii) List the set of all states (Q), the set of all input symbols ($\sum$), the initial state (q_0), and the set of final states (F) for this FSA.

[6 marks]

(b) Draw a finite state diagram that will accept the RE = M*P(F | G).

[4 marks]

Continued...

(c) For each of the Turing Machines (TMs) in Figure Q1(c)(i) and Figure Q1(c)(ii), write down the content of the tape and the position of the read/write head at the end of the execution when the machine stops.

[4 + 4 marks]

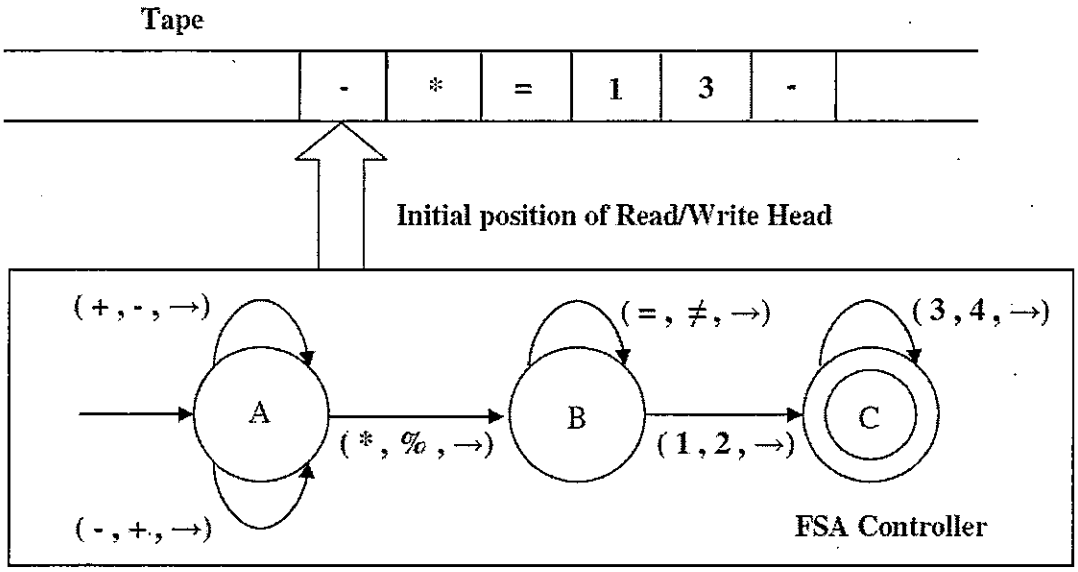

Figure Q1(c)(i)

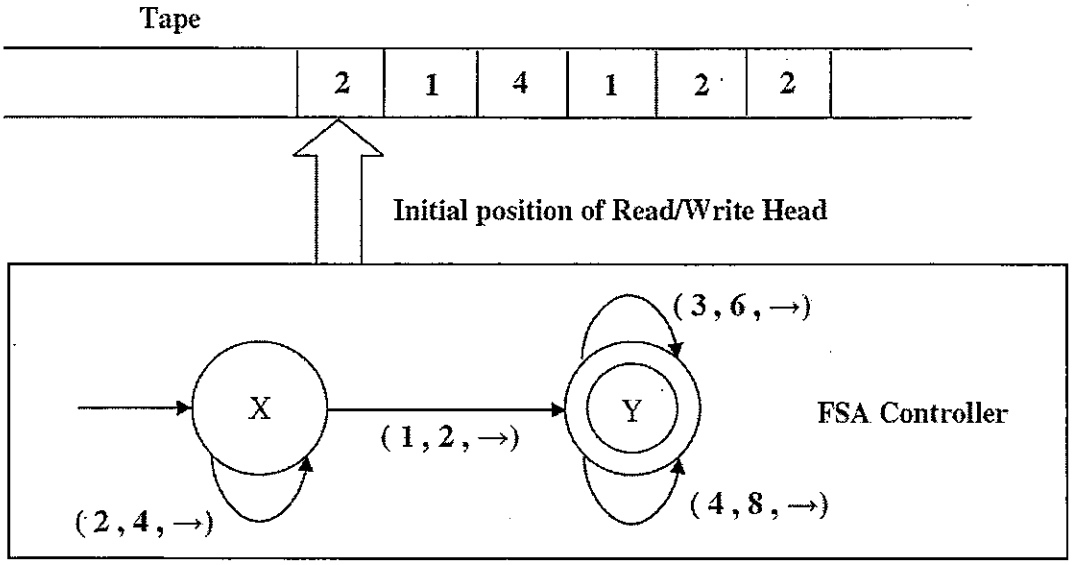

Figure Q1(c)(ii)

Continued...

Question 2

(a) An incomplete linked list program is given in Program Listing Q2(a)

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define N 10

//Answer for Part (a)(i)

struct Chess *newChess(char name[ ], int p21, int p22) {
   //Answer for Part (a)(ii)
};

int main() {
   int j = 0;
   char PlayerName[N];
   int p21, p22;
   struct Chess *head, *curr, *highest;

   //Answer for Part (a)(iii)

   //Answer for Part (a)(iv)

   Return 0;
}
```

Program Listing Q2(a)

Table Q2(a)

Chess Player Name	Total Points in 2021	Total Points in 2022
Amna	1832	2750
Azzam	2722	1922
Atiya	1540	2324
Aufa	2500	2800

(i) Define a **linked list node** structure named `Chess` that contains a chess player name and the points he or she collected for years 2021 and 2022 (see Table Q2(a))

[3 marks]

(ii) Write the function definition for `newChess` that will create and initialize a new `Chess` node.

[4 marks]

(iii) Write C statements to build a linked list to store data in Table Q2(a). Use the `newChess` function.

[4 marks]

(iv) Write C statements to **find** and print the player having the highest points in year 2021, as shown below:

```
Highest ranking in 2021 is Azzam, with points of 2722.
```

[4 marks]

Continued...

(b) Study the following C program in Fig. Q2(b).

```
#include <stdio.h>

void printSet(char name[], int s) {
    int j;
    printf("%s = { ",name);
    for(j=0;j<16; j++)
        if((1<<j)&s) printf("%d ",j);
    printf("}\n");
}

int main() {
    unsigned int setA = (1<<0)|(1<<2)|(1<<7); //part (i)
    unsigned int setB = (1<<1)|(1<<2)|(1<<7)|(1<<9); //part (i)

    //Answer for part (ii)

    printSet("Set A", setA); //part (iii)
    printSet("Set B", setB); //part (iii)

    //Answer for part (iv)

    return 0;
}
```

Figure Q2(b)

(i) Compute the initial decimal values of **`setA`** and **`setB`**.

[4 marks]

(ii) Write C code to insert element 4 into **`setB`**.

[2 marks]

(iii) Write C code to display the union and the intersection of **`setA`** and **`setB`**.

[4 marks]

Continued...

Question 3

(a) Consider the recursive formula:

$$F(n) = \begin{cases} 2 & n = 1 \\ G(n-1) * 2 & n > 1 \end{cases}$$

$$G(n) = \begin{cases} 3 & n = 1 \\ F(n-1) + 3 & n > 1 \end{cases}$$

(i) Write *two* mutually recursive functions in C programming language based on the formula above. Assume `n` is an integer greater or equal to `1`.

[6 marks]

(ii) Write a *complete* C program to test the recursive functions you wrote in part (a) (i). You need to include the recursive functions in your answer and show their positions relative to the other parts of the program. Print the values of `F(n)` and `G(n)` to the monitor for `n` = `1` to `5`, as shown in the example below (values shown are not the actual values of the functions).

```
F: 4 7 20 37 120
G: 1 9 33 56 140
```

[6 marks]

(iii) Predict the values of `F(n)` and `G(n)` for `n` = `2,3,4`.

[6 marks]

(b) The *K*-Knights problem is to place *K* Knights on an $M \times N$ chessboard in such a way that they do no attack each other.

(i) Describe the three steps of the backtracking algorithm used to solve the *K*-Knights problem.

[3 marks]

(ii) Figure Q3(b) shows a 3×3 chessboard with a Knight. Using the algorithm described in part (b) (i), complete the solution for the 5-Knights problem by placing the other 4 Knights onto the chessboard. Reproduce Figure Q3(b) in your answer, showing the positions of the 5 Knights.

[4 marks]

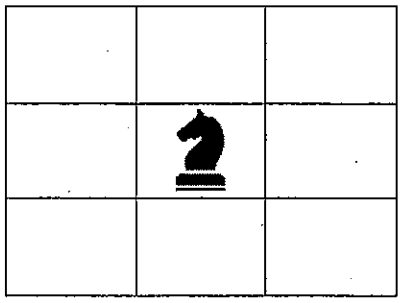

Figure Q3(b)

Continued...

Question 4

(a) Suggest a suitable *search* algorithm for each of the arrays: *Array P* and *Array Q*. Explain your answer.

[4 marks]

$$Array\ P = ["A" \quad "brown" \quad "fox" \quad "jumps" \quad "over"]$$

$$Array\ Q = ["Life" \quad "is" \quad "full" \quad "of" \quad "beauty"]$$

(b) The incomplete C program as shown in Program Listing Q4(b) implements a sort on an array of records. Study the code snippet carefully and answer the following questions.

```
#include <stdio.h>
#include <string.h>
#define SIZE 6

struct Cake {
   char name[50];
   float price;
};

int searchChoice(char *choice, struct Cake data[]);
void displayChoice(int index, struct Cake data[]);

int main() {
   char choice[50];
   int index;

   //Answer for Part (i) - Declare and initialise array stock
   printf("What is your choice of cake? ");
   gets(choice);

   index = searchChoice(choice, stock);
   displayChoice(index, stock);

   return 0;
}

//Answer for Part (ii) - Function definition for searchChoice

//Answer for Part (iii) - Function definition for displayChoice
```

Program Listing Q4(b)

Table Q4(b)

Cake	Price (RM)
Black Forest	12.50
Red Velvet	10.90
Chocolate Indulgence	11.90
Lemon Fudge	9.90

(i) Declare an array called `stock` of type `struct Cake` and initialize with the data in Table Q4(b)

[4 marks]

(ii) Write the function definition for `searchChoice` that implements a *sequential search* to search for a name of cake in an array of records. The function takes in the `choice` and array of `Cake` structures. If a match is found, the function returns the index of the matching element, otherwise it returns `-1`.

[5 marks]

(iii) Write the C function `displayChoice` that prints out the price of the selected cake. The function takes in the `index` and array of `Cake` structures. If the index equals -1, display "`Your choice is not available`" instead.

[4 marks]

(c) Sort by hand (no coding required) the word "`MALAYSIA`" in *ascending* order using the following sorting algorithms. Show all steps involved.

(i) Selection sort

[4 marks]

(ii) Insertion sort

[4 marks]

End of Paper